REMARKS/ARGUMENTS

The Examiner rejected claims 1-11, 13-25, 27-39, and 41-42 as anticipated (35 U.S.C. §102(e)) by Ireland (U.S. Patent No. 6,266,666). Applicants traverse.

Independent claims 1, 15, and 29 concern enabling access to data, and require: receiving a call from a client to invoke a remote interface method; accessing, with a remote interface implementation, parameters from the received call in response to the invocation of the remote interface method; generating a stored procedure call with the accessed parameters as input parameters of the stored procedure; transferring the stored procedure call to a stored procedure named by the call to execute; receiving output from the stored procedure; inserting the received output from the stored procedure into a data object; and returning the data object to the client.

The Examiner cited col. 9, lines 48-55 and col. 10, lines 45-49, and class ParamDef, col. 11 of Ireland as disclosing the claim requirement of generating a stored procedure call with the accessed parameters as input parameters of the stored procedure. (Office Action, pg. 3) Applicants traverse.

The cited col. 9 mentions that the user can invoke a window, such as a stored procedure window, for illustrating that a method on a component can be made to look like a stored procedure by returning a tabular result set. The cited col. 10 mentions implementing "methods as stored procedures". Although the cited cols. 9 and 10 discuss how a method on a component can be used to return a result set and look like a stored procedure, nowhere do the cited cols. 9-10 disclose generating a stored procedure call with the parameters accessed from the received call to the remote interface method.

In fact, the cited Ireland actually teaches away from the claim requirement generating a stored procedure call because Ireland uses a method on a component to obtain a result set that may look like from a stored procedure, but in fact is not the result of a stored procedure. Ireland mentions the component interface is called to return an object or a result set (collection of objects.) (Ireland, col. 3, lines 51-57, col. 4, lines 8-13) Thus, Ireland discusses calling a

method or interface on a component to return a result set. Nowhere does the cited Ireland actually disclose the claim requirement of generating a stored procedure call with parameters from a call to a remote interface. Instead, the call to the method of Ireland returns a result set. Ireland emphasizes that the described CTS "is done in an effort to 'mimic' an actual database server. Here the client thinks it is talking to a database server, but is actually talking to the CTS." (Ireland, col. 13, lines 33-43)

Moreover, Ireland mentions that after specifying argument information, the information is executed against the component residing in the middle tier, not the database. (Ireland, col. 10, lines 4-8) Thus, the cited Ireland executes the method against a component and does not mention generating a stored procedure call.

The Examiner cited col. 9, lines 51-60 as disclosing the claim requirement of transferring the stored procedure call to a stored procedure named by the call to execute. (Office Action, pg. 3) Applicants traverse.

The cited col. 9 mentions sending a query requesting a list of stored procedures and that a component transaction server enumerates components and methods on those components to make the component transactions appear as stored procedures. Nowhere does this cited col. 9 anywhere disclose the claim requirement of transferring the stored procedure call to a stored procedure named by the call to execute. Instead, the cited col. 9 discusses how components can be made to look like stored procedures. There is no disclosure or mention of transferring a generated stored procedure call to the named stored procedure to execute.

The Examiner cited col. 9, lines 47-51 as disclosing the claim requirement of receiving output from the stored procedure. (Office Action, pg. 3) Applicants traverse.

The cited col. 9 mentions a stored procedure window and how a method on a component can be made to look like a stored procedure to return a tabular result set. Nowhere does the cited col. 9 anywhere disclose receiving output from a stored procedure. Although the returned result set may look like a stored procedure result set, the cited col. 9 does not disclose that the result set is in fact returned by a stored procedure as claimed. In fact, Ireland discusses how the method is

executed against the component to return the result set, not executed and returned by a stored procedure as claimed. (Ireland, col. 10, lines 4-10) According to Ireland, the method is looked up and if found the component is created, and the method is called on the component to return the result set. (Ireland, col. 14, lines 5-15)

The Examiner cited col. 9, lines 47-51 as disclosing the claim requirement of inserting the received output from the stored procedure into a data object. (Office Action, pg. 3) Applicants traverse.

The cited col. 9 mentions a stored procedure window and how a method on a component can be made to look like a stored procedure to return a tabular result set. Nowhere does the cited col. 9 anywhere disclose that output from a stored procedure is added to a data object. In fact, as discussed above, the result set of the cited Ireland comes from a method called on the component, not from a stored procedure as claimed.

Accordingly, claims 1, 15, and 29 are patentable over the cited art because the cited Ireland does not disclose all the claim requirements.

Claims 2-11, 16-25, and 30-39 are patentable over the cited art because they depend from one of claims 1, 15, and 29. The following dependent claims provide additional grounds of patentability over the cited art for the reasons discussed below.

Claims 2, 16, and 30 depend from claims 1, 15, and 29 and further require that the stored procedure executes in a database server and generates the output, wherein the output is capable of comprising output that is a member of the set of output comprising one or more result sets of data from the database table and one or more output parameters resulting from stored procedure operations performed on data in the database table.

The Examiner cited col. 9, lines 47-55 and col. 14, lines 39-40 as disclosing the claim requirement that the that the stored procedure executes in a database server and generates the output. (Office Action, pg. 3) Applicants traverse.

The cited col. 9 discusses how a method on a component is called to look like a stored procedure so that returns the result set. The cited col. 14 mentions discusses how a client can

send a request to a server. Although the cited col. 9 discusses how the result may look like a stored procedure result set, nowhere does the cited Ireland disclose that the output result set is returned from a stored procedure executing in a database. In fact, Ireland appears to teach away from this requirement in its statement that "[h]ere, the client thinks its is talking to a database server, but is actually talking to the CTS". (Ireland, col. 13, lines 33-40).

Accordingly, claims 2, 16, and 30 provide additional grounds of patentability over the cited art because the cited Ireland does not disclose all the dependent claim requirements.

Claims 3, 17, and 31 depend from claims 1, 15, and 29 and further require that the remote interface implementation, an input mapping to determine the parameters in the client call to use as input parameters to the stored procedure call. The Examiner cited col. 12, lines 8-26, 54-59, 64-67, col. 13, lines 1-30, and col. 16, liens 62-67 as disclosing the claim requirements. (Office Action, pg. 4)

The cited col. 12 mentions class metadata that is used to load the components metadata into a hash table. The key for the hash table is a a scoped method name. Although the cited col. 12 discusses a hash table, nowhere does the cited col. 12 anywhere disclose or mention that the hash table is used to map parameters in the client call to input parameters in a stored procedure call as claimed.

The cited col. 13 mentions a table of commands. The cited col. 16 mentions invoking a method and calling a marshaller when a parameter is needed. Nowhere do these cited cols. 13 and 16 anywehre disclose or mention a mapping to map parameters in the client call to input parameters in a stored procedure call as claimed.

Accordingly, claims 3, 17, and 31 provide additional grounds of patentability over the cited art because the cited Ireland does not disclose all the dependent claim requirements.

Claims 5, 19, and 33 depend from claims 1, 15, and 29 and further require processing an output mapping indicating how the stored procedure output is mapped to the data object. The examiner cited col. 10, lines 36-42 of Ireland as disclosing the additional requirements of these claims. (Office Action, pg. 4) Applicants tarverse.

The cited col. 10 mentions generating a component and that for a Java component, a stub is generated that resembles a java class that JDBC for sending requests to the component transaction server for receiving result sets back.

Nowhere does the cited col. 10 anywhere disclose or mention the claim requirement of an output mapping to map stored procedure output to a database object. Instead, the cited col. 10 discusses generating a component graphically.

Accordingly, claims 5, 19, and 33 provide additional grounds of patentability over the cited art because the cited Ireland does not disclose all the dependent claim requirements.

Claims 6-12, 20-26, and 34-40 are patentable over the cited art because they depend from the base claims 1, 15, and 29 and provide additional details concerning the stored procedure, data object and other elements, which in combiantion with the base claims provide further grounds of patentability over the cited art.

Independent claims 13, 27, and 41 concern making stored procedure programs available to application programs, and require: determining one stored procedure program generating output needed by one application program; generating a remote interface implementation to respond to a remote interface method capable of receiving a call from the application program including data and invoking a stored procedure in a database server with the data from the application program used as input; and generating an output mapping for the remote interface implementation to use to determine how to insert the stored procedure output into a data object that may be used by the application program.

The Examiner cited blocks 210, 220, and 230 in FIG. 2 and col. 12, lines 29-61 as disclosing the claim requirement of generating a remote interface implementation to respond to a remote interface method capable of receiving a call from the application program including data and invoking a stored procedure in a database server with the data from the application program used as input. (Office Action, pg. 7) Applicants traverse.

The cited block 210 is a thin client, block 220 is a middle tier comprising the CTS, and block 230 is a back end database server. (FIG. 2, and col. 6, lines 3-15). The cited col. 12

mentions that a command class presents a request from a client. The command class is a collection of static methods. The runCommands() is called by the server's request handler when receiving a client request to determine the type of command.

Nowhere do the cited col. 12 and FIG. 2 anywhere disclose the claim requirements of generating a remote interface implementation to respond to a remote interface method capable of receiving a call from the application program including data and invoking a stored procedure in a database server with the data from the application program used as input. Nowhere does the cited Ireland anywhere disclose invoking a stored procedure in a database server in response to a call from the application program as claimed.

The Examiner cited col. 10, lines 13-20, 36-42 and col. 8, lines 16-31 as disclosing the claim requirement of generating an output mapping for the remote interface implementation to use to determine how to insert the stored procedure output into a data object that may be used by the application program. (Office Action, pg. 7)

The cited col. 10 mentions that the design allows a component to return a result back to the client, and provides an API for pushing a data set to clients, and interface calls are provided for describing every column, binding variables to those columns and then sending the data. On the client side, the user is able to generate a component graphically and for a Java component, the system generates a stub. The cited col. 8 mentions that the result set the CTS provides is a tabular result set and is equivalent to a database cursor. The CTS manages updates to the result set.

Nowhere do the cited cols. 8 and 10 disclose the claim requirement of generating a mapping for the remote interface to determine how to insert stored procedure output into a data object for use by the application program. Instead, the cited cols. 8 and 10 discuss returning a result set, but do not disclose mapping stored procedure output to a data object for use by an application program as claimed.

Accordingly, claims 13, 27, and 41 are patentable over the cited art because the cited Ireland does not disclose all the claim requirements.

Claims 14, 28, and 42 are patentable over the cited art because they depend from claims 13, 27, and 41, which are patentable over the cited art for the reasons discussed above.

The Examiner rejected claims 12, 26, and 40 as obvious (35 U.S.C. §103) over Ireland in view of Clegg (U.S. Patent No. 6,356,946). Applicants traverse and submit that these claims are patentable over the cited art because they depend from claims 1, 15, and 29, which are patentable over the cited art for the reason discussed above.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1-42 are patentable over the art of record. Applicants submit herewith a petition for a one-month extension of time and the accompanying fee. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0460.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: January 20, 2005

By: _____

David W. Victor
Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984